Datenbank-Spektrum

Das Fachgebiet „Informationssysteme" am Hasso-Plattner-Institut

--Manuscript Draft--

Manuscript Number:	
Full Title:	Das Fachgebiet „Informationssysteme" am Hasso-Plattner-Institut
Article Type:	Datenbankgruppen vorgestellt
Corresponding Author:	Felix Naumann Hasso-Plattner-Institut GERMANY
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Hasso-Plattner-Institut
Corresponding Author's Secondary Institution:	
First Author:	Felix Naumann
First Author Secondary Information:	
Order of Authors:	Felix Naumann
	Ralf Krestel
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	Das Hasso-Plattner-Institut (HPI) ist ein privat finanziertes Institut an der Universität Potsdam. Stifter ist Professor Hasso Plattner, Mitgründer und Aufsichtsrats-vorsitzender des Software-Konzerns SAP. Das Fachgebiet Informationssysteme, das von Prof. Dr. Felix Naumann geleitet wird, beschäftigt sich mit dem effizienten und effektiven Umgang mit heterogenen Daten und Texten. Gegründet wurde das Fachgebiet 2006 und bietet derzeit 12 Doktoranden und circa 15 Masterstudenten eine Forschungsumgebung.
Suggested Reviewers:	

Das Fachgebiet „Informationssysteme" am Hasso-Plattner-Institut

Felix Naumann und Ralf Krestel
Hasso-Plattner-Institut
14482 Potsdam
vorname.nachname@hpi.de

Zusammenfassung

Das Hasso-Plattner-Institut (HPI) ist ein privat finanziertes Institut an der Universität Potsdam. Stifter ist Professor Hasso Plattner, Mitgründer und Aufsichtsratsvorsitzender des Software-Konzerns SAP. Das Fachgebiet Informationssysteme, das von Prof. Dr. Felix Naumann geleitet wird, beschäftigt sich mit dem effizienten und effektiven Umgang mit heterogenen Daten und Texten. Gegründet wurde das Fachgebiet 2006 und bietet derzeit 12 Doktoranden und circa 15 Masterstudenten eine Forschungsumgebung.

1. Motivation

Daten sind in großer Fülle vorhanden – man findet sie in vielen verschiedenen Formen von herkömmlichen relationalen oder XML-Datenbanken über semi-strukturierte Daten, oft verlinkt und veröffentlicht als Open Data, bis hin zu Textdaten aus Webdokumenten. Diese Fülle an Daten wird immer größer, und viele Organisationen und Wissenschaftler haben die Vorteile erkannt, diese Daten zu größeren, homogeneren, konsistenteren und saubereren Mengen zu integrieren. Datenintegration vereint unzusammenhängende Quellen in Organisationen; sie bietet den Konsumenten außerdem eine vervollständigte Sicht auf Produktangebote; die Kombination experimenteller Ergebnisse führt zu Gewinnung von neuen wissenschaftlichen Erkenntnissen.

Jedoch ist diese Art von Integration aufgrund der vielfältigen Heterogenitäten schwierig. Syntaktische Heterogenität in Datenformaten, Zugriffsprotokollen und Anfragesprachen ist recht einfach zu lösen, gewöhnlich durch das Bilden von geeigneten quellenspezifischen Wrapper-Komponenten. Des Weiteren muss die strukturelle Heterogenität überwunden werden, indem man verschiedene Schemata aneinander ausrichtet. Sogenannte Schema-Matching-Techniken liefern automatisch Ähnlichkeiten und Korrespondenzen entlang der Schemaelemente, während Schema-Mapping-Techniken die Korrespondenzen als eigentliche Datentransformationen interpretieren. Um schließlich auch die semantische Heterogenität zu überwinden, müssen die unterschiedlichen Bedeutungen von Daten und die ähnliche aber doch unterschiedliche Repräsentation von Echtwelt-Entitäten erkannt werden. Hierfür werden Ähnlichkeitssuche und Datenbereinigungstechniken eingesetzt.

Während wir uns den ersten beiden Herausforderungen in der Vergangenheit gewidmet haben, ist die letzte und wohl auch schwerste, diejenige, auf die wir den Hauptfokus unserer Forschungsarbeit legen. Dieser Fokus lässt sich in drei Hauptforschungsrichtungen aufteilen, welche in den folgenden Kapiteln beschrieben werden: Unser erstes und jüngstes Gebiet ist das Data Profiling, also die Entwicklung von Methoden, um interessante Eigenschaften über unbekannte Datenmengen zu entdecken. Unsere zweite Ausrichtung ist das Gebiet der Datenbereinigung (Data Cleansing), also der Entwicklung von Methoden, um automatisiert Fehler und Unregelmäßigkeiten in Datenbanken zu beheben, insbesondere, um Duplikate zu suchen und zu konsolidieren. Die dritte Forschungsrichtung ist das Text Mining, also das Extrahieren von Informationen aus Textdaten, wie zum Beispiel Wikipedia-Artikel, Tweets und andere Texte aus dem Web.

Soweit es uns möglich ist, versuchen wir unsere Daten und Algorithmen frei zugänglich bereitzustellen: http://hpi.de/naumann/projects/repeatability.html.

2. Data Profiling

„*Data Profiling* ist eine Folge von Maßnahmen und Prozessen, um Metadaten über eine gegebene Datenmenge zu bestimmen" [1]. Die Notwendigkeit neue und (noch) unbekannte Daten initial zu untersuchen, wird in vielen Situationen offenkundig, typischerweise in der Vorbereitung auf Folgeaufgaben. Das Profiling umfasst eine breite Palette an Methoden, um effizient Datenmengen zu analysieren. In einem typischen Szenario, welches die

Fähigkeiten kommerzieller Data-Profiling-Werkzeuge widerspiegelt, werden Tabellen aus relationalen Datenbanken untersucht, um Metadaten zu ermitteln. Dazu gehören Datentyp und typische Wertmuster, Vollständigkeit sowie Einzigartigkeit von Spalten, Schlüssel und Fremdschlüssel, und gelegentlich auch funktionale Abhängigkeiten sowie Assoziationsregeln. Zusätzlich hat die Forschung (sowohl unsere eigene als auch andere) Ansätze für die Entdeckung weiterer Metadaten hervorgebracht, wie z.B. die der Entdeckung von Inklusionsabhängigkeiten oder bedingten funktionalen Abhängigkeiten. Es gibt eine Vielzahl an konkreten Nutzungsbeispielen für die Ergebnisse des Profiling:

- **Anfrageoptimierung:** Anzahlen und Histogramme für die Selektivitätsschätzung, Abhängigkeiten für Anfragevereinfachung
- **Datenbereinigung:** Muster- und Abhängigkeitserkennung, um dann Verstöße festzustellen
- **Datenintegration:** Inklusionsabhängigkeiten über Datenbanken hinweg, um Daten anzureichern und Join-Pfade zu finden
- **Datenanalyse:** Datenaufbereitung und erste Einblicke gewinnen
- **Rekonstruktion von Datenbanken:** Entdeckung von Fremdschlüsseln, um Schemata zu verstehen und deren wichtigsten Komponenten zu identifizieren

Unser Überblicksartikel zeigt den Fortschritt der Community in diesem Bereich [1]. Data Profiling erfreut sich einer immer größeren Aufmerksamkeit, da Forscher und Praktiker zunehmend erkennen, dass das bloße Zusammentragen von Daten in einen großen See (Data Lake) nicht ausreichend ist. „Wenn wir bloß einen Haufen an Datensätzen in einer Ablage sammeln, ist es unwahrscheinlich, dass jemals jemand in der Lage sein wird, diese aufzufinden, geschweige denn diese nochmalig zu verwenden. Mit angemessenen Metadaten besteht jedoch Hoffnung […]" [4].

2.1 Profiling relationaler Daten

Abgesehen von den weniger komplexen Aufgaben, wie das Ermitteln der Anzahl unterschiedlicher Werte in einer Spalte, betrachtet Data Profiling komplexe Aufgaben, wie zum Beispiel alle Abhängigkeiten in einer großen Datenmenge zu ermitteln. Wir und auch andere Forschungsgruppen haben verschiedenartige Methoden zur effizienten Entdeckung aller minimalen funktionalen Abhängigkeiten, Inklusionsabhängigkeiten, eindeutiger Spaltenkombinationen und Ordnungsabhängigkeiten entwickelt. Weitere Abhängigkeiten sollen folgen, beispielsweise Join-Abhängigkeiten, Matching-Abhängigkeiten oder sogenannte Denial Constraints. Anstatt nun jede Profiling-Aufgabe detailliert zu beschreiben, heben wir die allgemeinen Schwierigkeiten hervor, die Data Profiling besonders herausfordernd, aber auch spannend machen:

Schemagröße: Da Abhängigkeiten in jeglichen Spalten und auch Spaltenkombinationen auftreten können, ist nicht nur die Anzahl der Datensätze, sondern insbesondere die Anzahl der Spalten ein ausschlaggebender Faktor für die Problemkomplexität.

Größe der Abhängigkeiten: Eine Möglichkeit, den exponentiellen Suchraum zu beschränken, ist die Größe der Abhängigkeiten zu limitieren, also die Anzahl der beteiligten Attribute. Zum Beispiel könnte man festlegen, dass Schlüsselkandidaten mit mehr als zehn Attributen irrelevant sind. Anderseits kann aber die vollständige Metadaten-Menge nützlich sein, um beispielsweise eine Relation auf der Basis ihrer funktionalen Anhängigkeiten zu normalisieren [18].

Anzahl der Abhängigkeiten: Während die meisten abhängigkeitsfokussierten Arbeiten, wie etwa Normalisierungstheorie oder das Schließen über Abhängigkeiten, von einer Handvoll an Abhängigkeiten als Input ausgeht, haben wir es üblicherweise mit Tausenden, Millionen oder sogar Milliarden an Abhängigkeiten in echten Datenbanken zu tun. So wird schon deren bloße Speicherung zu einem Problem, ganz zu schweigen von logischem Schlussfolgern oder einer Interpretation durch einen Experten.

Behandlung von Nullwerten: Die Semantik fehlender Werte ist ein spannendes Problem für jede Datenmanagement- und Analyseaufgabe, was auch auf Data Profiling zutrifft [11].

Komplexes Pruning: Huhtala et al. haben schon vor Längerem recht komplexe Regeln aufgezeigt, die den Suchraum für die Entdeckung funktionaler Abhängigkeiten beschneiden [10]. Wenn man Profiling für verschiedene Arten von Abhängigkeiten betreibt, wird es möglich, auch über Abhängigkeiten hinweg den Suchraum zu beschneiden.

Relaxierte Abhängigkeiten: Abgesehen von strikten Abhängigkeiten ist es auch von Interesse, partielle Abhängigkeiten zu entdecken, also solche, die nur für einem bestimmten Teil der Datenbank wahr sind, und bedingte Abhängigkeiten (*Conditional Dependencies*), die in einem wohl-definierten Bereich wahr sind.

Dynamische Daten: Obwohl unser Fokus auf Algorithmen für statische Datenmengen liegt, sind wir auch daran interessiert, Metadaten für sich verändernde Daten effizient aktuell zu halten, ohne stets alles erneut zu ermitteln.

Experimente: Das Prüfen der Korrektheit von Algorithmen für gegebene Echtweltdaten ist recht unkompliziert. Hingegen gestaltet sich das Generieren synthetischer Testdaten mit speziellen Eigenschaften, wie zum Beispiel eine bestimmte Anzahl und Verteilung an funktionalen Abhängigkeiten, sehr schwierig.

Ergebnisse interpretieren: Entdeckte Metadaten können nur für die aktuelle Instanz validiert werden. Manche Abhängigkeiten werden auch im Allgemeinen zutreffen, andere wiederum nicht. Auf diese wohl wichtigste und dabei schwierigste Herausforderung der Interpretation und Nutzung von Data-Profiling-Ergebnissen gehen wir in Abschnitt 2.4 ein.

Abschließend kann man feststellen, dass Data Profiling weiter viele offene Forschungsfragen anbietet!

2.2 Das Metanome-Projekt

Metanome ist unser Java-gestütztes Framework und Werkzeug um relationale Datensätze und Data-Profiling-Algorithmen zu verwalten [16]. Unsere Motivation ist es, die vielen in unserer Gruppe entwickelten und aus anderen Arbeiten nachimplementierten Algorithmen zu bündeln und eine einfache Schnittstelle und Testumgebung für Entwickler neuer Algorithmen zu schaffen. So ermöglichen wir letztlich faire Vergleiche unter konkurrierenden Algorithmen. Unser erstes Augenmerk galt der Entdeckung funktionaler Abhängigkeiten – in Metanome sind mittlerweile acht veröffentlichte FD-Entdeckungsalgorithmen implementiert, inklusive derer, die in [17] ausgewertet werden. Es kommen noch acht weitere Algorithmen für andere Profiling-Aufgaben hinzu (www.metanome.de).

2.3 RDF Profiling

Von besonderem Interesse, da reichhaltig, vielfältig und oft allgemein verständlich, sind Datensätze aus dem Bereich Linked (Open) Data. Um diese zu untersuchen, wenden wir sowohl herkömmliche, als auch neuartige Data-Mining-Technologien für Linked Data in deren RDF-Darstellung als Subjekt-Prädikat-Objekt-Tripel an. Beispielsweise erlaubt uns die Entdeckung häufiger Kombinationen (*Frequent Itemsets*) von Eigenschaften (Prädikaten) oder Objekten im Kontext von Subjekten, Daten mit fehlenden Tripeln anzureichern. Eine weitere Konfiguration – häufige Subjekte im Kontext von Eigenschaften – erlaubt das Clustering von Entitäten. Des Weiteren haben wir Data-Mining-Technologien für die Entdeckung von bedingten Inklusionsabhängigkeiten eingesetzt [14]. Der typische Umfang solcher Daten (ein populärer Datensatz stammt aus der Billion-Tripels-Challenge und enthält gegenwärtig über drei Milliarden Tripel) erfordert besonders Speicher-effiziente Algorithmen.

Die meisten Ergebnisse dieser Arbeit fließen in unser webbasiertes Werkzeug ProLOD++ ein, welches Methoden anbietet, um in RDF-Daten Schlüsselkandidaten zu entdecken, Klassen- und Eigenschaftsverteilungen zu erforschen, häufige grafische Muster zu entdecken und vieles mehr (siehe [2] und Abbildung 1).

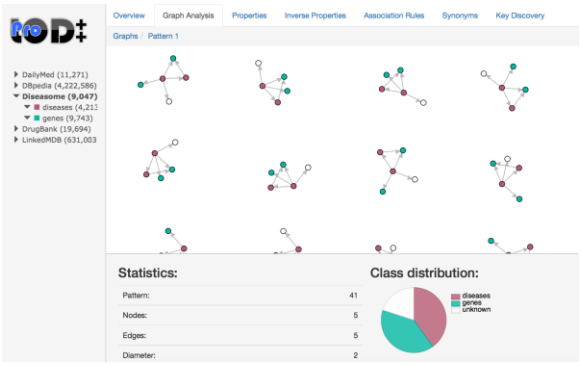

Abbildung 1: Erforschung häufiger Muster im verknüpften Datensatz mit ProLOD++ (www.prolod.org)

2.4 Von Metadaten zu Semantik

Das Finden aller (und somit meist sehr vieler) Abhängigkeiten in einer gegebenen Datenmenge ist erst der Anfang einer sinnvollen Analyse. Die überwiegende Mehrheit an ermittelten Metadaten ist meist falsch: Sie mögen nur in der aktuellen Instanz gültig sein, oder aber durchaus in allen anderen möglichen Instanzen gültig, aber dennoch bedeutungslos sein. Die Spreu vom Weizen zu trennen ist sehr

schwierig, da es den Sprung von (Meta-)Daten zu Semantik darstellt; nur ein Mensch kann einen Schlüsselkandidaten zum Schlüssel erheben, Inklusionsabhängigkeiten zu Fremdschlüsseln machen oder funktionale Abhängigkeiten zu einer Nebenbedingung befördern.

Doch die Informatik kann helfen: Wir verbringen derzeit einen Großteil unserer Zeit, große Mengen an Metadaten in Schema-Informationen umzuwandeln. Der erste Schritt dabei ist die Entwicklung eines Metadaten-Management-Systems, das Metadaten speichert. Als nächstes entwickeln wir Auswahl- und Rankingmethoden, um Nutzern nur die vielversprechendsten Metadaten zu präsentieren. Abschließend stellt die Visualisierung von Metadaten ein wichtiges Werkzeug dar, um Experten Hilfestellung zu leisten und sie ihre Daten besser verstehen zu lassen. Abbildung 2 zeigt beispielsweise Zusammenhangskomponenten, die durch die Entdeckung von Inklusionsabhängigkeiten in Millionen von Web-Tabellen entstanden sind.

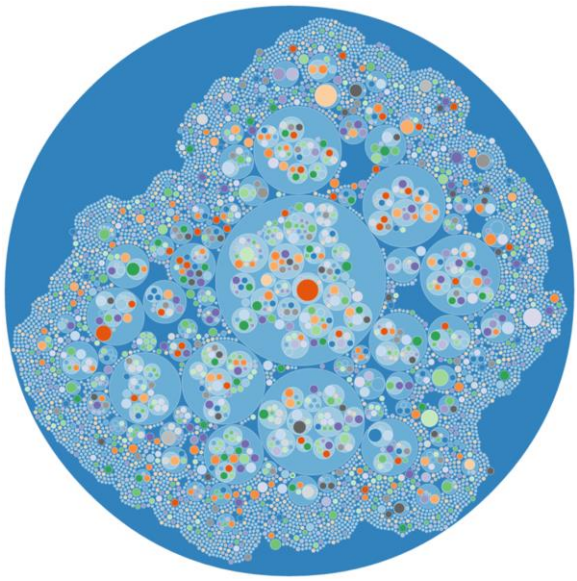

Abbildung 2: Cluster aus Web-Tabellen, verbunden durch (sinnvolle) Inklusionsabhängigkeiten

3. Datenbereinigung

Mit wachsenden Datenmengen wachsen auch Probleme mit der Datenqualität. Eines der interessantesten Probleme ist die mehrfache, jedoch unterschiedliche Datenrepräsentation des gleichen Objektes – so genannte Duplikate. Diese Duplikate haben gleich mehrere negative Auswirkungen: Zum Beispiel können Bankkunden doppelte Identitäten erhalten, Lagerbestände werden falsch überwacht, Kataloge werden mehrfach an die gleichen Haushalte geliefert, etc. Ein ähnliches Problem ist das der Ähnlichkeitssuche in strukturierten Daten: Zu einer Suchanfrage in Form eines Datensatzes sollen die ähnlichsten Datensätze in einer Datenbank gefunden werden und es soll entschieden werden, ob einer davon eine Übereinstimmung aufweist.

Beide Bereiche, Ähnlichkeitssuche und Duplikaterkennung, erfahren zurzeit eine Renaissance in Forschung und Industrie. Neben der Erarbeitung wissenschaftlicher Beiträge kooperieren wir mit Unternehmen zum Technologietransfer. Sowohl unsere Ähnlichkeitssuche als auch unsere Duplikaterkennungstechniken werden aktiv von Industriepartnern genutzt.

3.1 Duplikaterkennung

Das Aufspüren von Duplikaten in einer gegebenen Tabelle ist schwierig: Erstens sind die Darstellungen der Duplikate normalerweise eben nicht identisch, sondern deren Werte weichen etwas voneinander ab. Zweitens müssten im Allgemeinen alle Datensatzpaare verglichen werden, was für große Datenmengen undurchführbar ist. Unsere Forschung untersucht beide Aspekte, erstens durch den Entwurf effektiver Ähnlichkeitsmaße und zweitens durch das Entwickeln effizienter Algorithmen zur Reduzierung des Suchraumes.

Ein Fokus unserer Arbeit ist es, verbesserte Variationen der eleganten und einfachen Sorted Neighborhood-Methode zu entwickeln [9], zum Beispiel durch Anpassung an verschachtelte XML-Daten, durch progressive Verarbeitung, die möglichst früh möglichst viele Ergebnisse liefert, durch Parallelisierung für GPU-Verarbeitung oder durch eine adaptive Version, die beweisbar effizienter ist als das Original [5].

Aus unserer Erfahrung heraus leiden die meisten Projekte zum Thema Duplikaterkennung bei dem Versuch, Technologie und Methodik in die industrielle Wirklichkeit zu transferieren. Die Verfügbarkeit von Daten ist ein erstes Problem, welches aufkommt, selbst wenn eine Kooperation fest etabliert ist und alle teilnehmenden Parteien grundsätzlich dem Projekt zustimmen. Als nächstes werden Domänen- und Partner-spezifische Ähnlichkeitsmaße benötigt, die zum

jeweiligen Anwendungsfall passen. Unternehmen haben oftmals sehr unterschiedliche Sichtweisen darauf, was ein Duplikat überhaupt ist: Das Messen von *Recall*, also der Vollständigkeit der entdeckten Duplikate, ist unmöglich, in Anbetracht der großen Datenmengen und *Precision*, also der Korrektheit der entdeckten Duplikate, ist überraschend dehnbar, je nachdem, wen man zur Validierung befragt. Abschließend ist zu sagen, dass die reale Welt viele praktische Details vorhält, die in der Forschung bequem ignoriert werden können. Mit [20] gelang es uns, solche Schwierigkeiten zu überwinden und die Datenqualität unserer Partner zu verbessern.

3.2 Ähnlichkeitssuche

Ein Problem, das mit der Duplikatsuche zwar verwandt ist, jedoch abweichende Anforderungen hat, ist die effiziente Ähnlichkeitssuche. Anstatt alle oder zumindest sehr viele Datensatzpaare offline zu vergleichen (n×n), fragt die Ähnlichkeitssuche online nach allen Datensätzen, die zu einem gegebenen Anfragedatensatz passen (1×n). Ein typisches Fallbeispiel ist ein Callcenter-Mitarbeiter, der Kundendaten nur auf Basis des Kundennamens und Wohn-orts aus dem System auslesen möchte. Die größte Herausforderung ist dabei die Entwicklung eines passenden Ähnlichkeits-Index, eine weitaus komplexere Datenstruktur als die üblichen Indizes, die auf Gleichheit von Werten beruhen.

Eine unserer Lösungen basiert auf klassischen Ansätzen der Anfrageoptimierung: Wir wählen Ähnlichkeitsindexzugriffe aus, basierend auf ihrer Selektivität und ihrer Kosten, die jeweils durch einen dynamisch gewählten Schwellwert modifiziert werden: Ein niedriger Schwellwert ergibt mehr Kandidaten, bedeutet aber auch mehr Zugriffe und somit höhere Kosten, um die Kandidaten abzurufen und abschließend zu vergleichen [15]. Eine weitere Erkenntnis ist die Bedeutung von Häufigkeiten für Ähnlichkeitsmaße: Je nach Häufigkeit der Anfragewerte sollten unterschiedliche Gewichtungen vorgenommen werden (Leutheusser-Schnarrenberger vs. Müller).

Derzeit erweitern wir diese Ideen, um das Problem eines ständig wachsenden Datensatzes zu lösen, der Duplikat-frei gehalten werden soll: Jede Anfrage ist zugleich ein Einfügen in den Datenbestand.

4. TEXT MINING

Unstrukturierte Daten in Form von natürlichsprachlichen Textdokumenten findet man überall, wo Kommunikation und Informationsaustausch zwischen Menschen stattfindet. Entsprechend vielfältig sind die Dokumentarten. Insbesondere der Erfolg des Internets als Kommunikationsmedium hat zu einer Explosion öffentlich zugänglicher Textsammlungen geführt. Neben benutzergeneriertem Inhalt, wie beispielsweise Wikipedia, Blogs oder Tweets, gibt es auch eine Menge professionell erstellter Inhalte, wie zum Beispiel Patente, Zeitungsartikel, politische Reden, Romane oder wissenschaftliche Publikationen. Weniger frei zugänglich, aber auch in großen Mengen vorhanden, sind natürlichsprachliche Dokumente wie Patientenakten, Geschäfts-Emails oder Instant-Messaging-Nachrichten. Für das Text Mining stellen diese sehr heterogenen Daten eine besondere Herausforderung dar, da jedes Genre seine eigenen Charakteristiken besitzt, und Dokumente von wenigen Worten bis tausende von Seiten lang sein können. Um diese vielfältigen Daten zu analysieren, Muster zu erkennen und Wissen zu generieren, bedient sich das Text Mining klassischer Methoden des Information Retrievals, der automatische Sprachverarbeitung, der Informationsextraktion und des maschinellen Lernens. Aktuelle Projekte am Fachgebiet sind die Analyse von Nachrichten und von Geschäftskommunikation, die Erforschung und Weiterentwicklung von Topic-Modellen, die gemeinsame Analyse mehrerer unterschiedlicher Textsammlungen, sowie die Erforschung des zeitlichen Aspekts bei dynamischen, sich schnell ändernden Textsammlungen.

4.1 Analyse von Nachrichten

Nachrichten dienen als Grundlage mehrerer Text-Mining-Aufgaben, wie beispielsweise der automatischen Erzeugung von Zusammenfassungen oder dem Erkennen von Ereignissen. Uns interessieren Zeitungsartikel im Zusammenhang mit Medien-Bias. Um diesen zu erkennen, genügt die Analyse eines einzelnen Zeitungsartikels nicht, und meist auch nicht ein isolierter Blick auf alle Artikel einer Zeitung. Wir bedienen uns daher zusätzlicher Informationsquellen, und zwar parlamentarischer Reden sowie Leserkommentaren.

Vergleiche von parlamentarischen Reden mit Nachrichtenartikeln verschiedener deutscher Nachrichtenagenturen haben in ersten Experimenten gezeigt, dass der wahrgenommene Bias automatisch quantifiziert werden kann [12]. Das Benutzen eines bestimmten Vokabulars ist ein erster Indikator für Bias (z.B. „Kernenergie" vs. „Atomenergie" in Deutschland). Neben bestimmten Standpunkten (Statement Bias) können Zeitungen ihre Leser beeinflussen, indem sie nur über bestimmte Themen berichten (Gate-Keeping Bias) oder indem sie bestimmte Positionen stärker als andere abdecken (Coverage Bias). Die automatische Erkennung aller drei Bias-Arten ist besonders schwierig, da in der deutschen Medienlandschaft nur subtile Unterschiede zwischen den Leitmedien bestehen. Das macht es notwendig, nicht nur Entitäten (Politiker, Parteien, Experten) und ihre Beziehungen zu extrahieren und zu analysieren, sondern ebenso detaillierte Opinion-Mining und Sentiment-Analyse zu betreiben und alles ins Verhältnis zu einer virtuellen, objektiven Berichterstattung zu setzen.

Erste Ergebnisse konnten wir auch durch die Analyse von Zeitungsartikelkommentaren erzielen. Unsere Hoffnung war, dass Leser in ihren Kommentaren den Bias der Zeitung unverblümter offenbaren [6], was wir auch anhand einer Klassifizierungsaufgabe bestätigen konnten. Ein weiterer Schwerpunkt liegt auf dem Erkennen von Hasskommentaren, um im Idealfall die Veröffentlichung ebendieser automatisch zu unterbinden. Dafür arbeiten wir mit einer großen deutschen Tageszeitung zusammen.

4.2 Analyse von Geschäftskommunikation

Das Erkennen von Entitäten, insbesondere Firmennamen, und Themen in E-Mails ist der erste Schritt, um komplexe Zusammenhänge innerhalb eines Unternehmens und dessen Kunden zu erkennen. Im Rahmen eines Industrieprojektes mit einer großen deutschen Bank streben wir den Aufbau von Unternehmensnetzwerken zur Unterstützung ihrer Risikomanagementabteilung an. Diese Firmennetzwerke werden automatisch aus Zeitungsartikeln extrahiert und stellen eine neue Herausforderung für die benannte Entitätserkennungsaufgabe (Named Entity Recognition) dar, die aufgrund komplexer, oft zweideutiger Benennung besonders schwierig für deutsche Firmennamen ist. Darüber hinaus unterscheiden sich die Beziehungstypen, die für uns interessant sind, von üblichen, binären Beziehungen, wie „verheiratet mit" oder „wohnhaft in". Unsere Unternehmensnetzwerke erfordern die Erfassung von Beziehungen, die nicht notwendigerweise binär sein müssen, z.B. „Wettbewerber mit" oder „Lieferant für". Zu diesem Zweck haben wir einen Algorithmus entwickelt, welcher anhand weniger vorgegebener Instanziierungen einer Relation andere, gleichartige Beziehungen findet. Der Algorithmus basiert auf Snowball [3] und kann mit jeder Art von Beziehung, die vom Benutzer bereitgestellt wird, umgehen.

4.3 Topic-Modelle

Topic-Modelle sind statistische Modelle, welche in großen Dokumentsammlungen Wörter thematisch in sogenannten Topics gruppieren und damit einen Überblick über die vorhandenen Themen in einer Textsammlung liefern. Neben dem Einsatz dieser Modelle für unsere Analysen arbeiten wir auch an der Weiterentwicklung und Anpassung dieser Modelle für diverse Anwendungsszenarien. So haben wir beispielsweise die teilweise vorhandenen Schlagwörter bei Projektanträgen benutzt, um Projekte in Themenbereiche einzuordnen [19]. Dies hat auch die Möglichkeit eröffnet, die Summen der geförderten Projekte auf Themen umzulegen und Trends in der Gesundheitsforschungsförderung offenzulegen.

Ein weiterer Schwerpunkt liegt auf dem Einbinden sogenannter Word Embeddings zur Verbesserung von Topic-Modellen. Word Embeddings erlauben die Repräsentation von Wörtern in n-dimensionalen, reellen Vektorräumen und somit das Finden von semantisch ähnlichen Wörtern. Diese Technologie, welche auf Deep Learning beruht, machen wir uns zunutze, um allgemeine Wörter durch thematische relevantere Wörter in den Topics zu ersetzen.

4.4 Analyse mehrerer Textsammlungen

Ein besonderes theoretisches Problem, mit dem wir uns beschäftigen, ist die Analyse von Textdaten über Korpusgrenzen hinweg, was schon bei Zeitungsartikeln und Parlamentsreden nötig war. Hier haben wir eine Reihe praktischer Anwendungen untersucht, um systematisch Schwierigkeiten zu erkunden.

In einem Versuch, die Kluft zwischen traditionellen Nachrichten und Social Media zu

überbrücken, haben wir ein Tweet-Empfehlungs-System entwickelt [13]. Das Ziel war es, dem Leser eines News-Artikels über ein bestimmtes Ereignis einen Überblick über Reaktionen auf Twitter zu geben. Während Twitter häufig genutzt wird, um Informationen zu teilen und zu verbreiten, wird es ebenso häufig genutzt, um Meinungen auszudrücken, Ideen abzulehnen oder bestimmte Standpunkte zu unterstützen. Um diese Meinungen zu erkennen, müssen traditionelle Sentimentanalysetechniken angepasst werden, um Emojis, Abkürzungen, Slang usw. zu erkennen. Das Missverhältnis zwischen der Sprache, die in Nachrichtenartikeln und Tweets verwendet wird, macht die Empfehlung des einen aufgrund des anderen sehr schwierig.

4.5 Analyse dynamischer Korpora

Viele der analysierten Textsammlungen sind nicht statisch, sondern ändern sich mehr oder weniger schnell. Die Einbeziehung der zeitlichen Komponente bei der Analyse, aber auch bei konkreten Aufgaben, wie dem Empfehlen von Objekten, muss deshalb besonders Rechnung getragen werden. Twitter stellt durch die kurzen Textbeiträge und der hohen Frequenz an neuen Tweets eine besondere Herausforderung dar. Durch verschiedene zeitliche Modelle haben wir versucht, die Dynamik von Hashtags zu analysieren und die Ergebnisse zu verwenden, um das Empfehlen von Hashtags zu verbessern [8].

Weiter haben wir die längerfristigen, thematischen Veränderungen in einem Forum untersucht [7]. Hierbei konnten nicht nur unterschiedliche Softwareversionen anhand der zeitlichen Diskussion identifiziert werden, sondern auch der Wandel bei der Benutzung von Begriffen und der sich ändernde Kontext für bestimmte Wörter.

5. Lehre am Fachgebiet

Im Bachelorstudium bieten wir, neben den typischen Vorlesungen zu den Grundlagen und der Implementierung von Datenbankmanagementsystemen und den sporadischen Proseminaren, jährlich ein bis zwei Bachelorprojekte an – eine besondere Veranstaltungsform am HPI. Zwischen vier und acht Bachelorstudenten bearbeiten zum Abschluss ihres Studiums ein Softwareprojekt über einen Zeitraum von zwei Semestern und stets in Kooperation mit einem externen Partner, der die Aufgabenstellung vorgibt und gleichsam als „Kunde" das Projekt aktiv begleitet. Die individuellen Bachelorarbeiten greifen Themen des Bachelorprojekts auf. Die Studenten sind im Wintersemester circa zur Hälfte ihrer Zeit im Rahmen des Projekts tätig, im darauf folgenden Sommersemester bearbeiteten sie in der Regel ausschließlich die Aufgaben im Projekt. Aus der Studienordnung: „Es handelt sich um praxisnahe Projekte, bei denen die Studierenden nicht nur als Entwickler kreativ werden, sondern in denen sie auch die besonderen Merkmale der Koordination von vielen Projektbeteiligten erleben." Die entstandenen Softwarelösungen werden zum Semesterabschluss der Öffentlichkeit vorgestellt und nicht selten anschließend durch die Partnerorganisationen produktiv eingesetzt. Unsere bisherigen Projektpartner waren u.a. Wikimedia Deutschland, Commerzbank, IBM, Capgemini sowie diverse kleine und mittelständige Unternehmen (https://hpi.de/naumann/teaching/bachelorprojekte.html).

Im Masterstudium bieten wir Vorlesungen und Seminare an, deren Themen durch unsere Forschungsrichtungen bestimmt werden. Zu den wichtigsten und wiederkehrenden Vorlesungen gehören „Information Integration", „Data Profiling", „Data Mining and Probabilistic Reasoning" sowie „Information Retrieval and Web Search". Seminare ergänzen den Stoff und werden des Öfteren als Projektseminare durchgeführt, bei denen neben den üblichen Vorträgen und Ausarbeitungen auch vergleichende Implementierungen erstellt werden. Auch im Masterstudium ist eine Teamarbeit vorgesehen, nämlich ein Masterprojekt mit drei bis sechs Studenten, die im Gegensatz zu den Bachelorprojekten eine konkrete Forschungsfrage des Fachgebiets untersuchen sollen. Ein typisches Ergebnis eines solchen Projekts ist die (oft erfolgreiche) Einreichung eines Manuskripts auf einer wissenschaftlichen Konferenz (https://hpi.de/naumann/teaching/master-projects.html).

Danksagung Unserer Forschung genoss die Unterstützung verschiedener Partner wie der DFG und Unternehmen, die sich für das Verständnis und die Verbesserung ihrer Daten interessieren. Die hier vorgestellten Arbeiten beruhen – natürlich – auf der Forschung unserer hervorragenden Doktoranden: Tobias Bleifuß, Toni Grütze, Hazar Harmouch, Maximilian Jenders, Anja Jentzsch, John Koumarelas, Sebastian Kruse, Konstantina Lazaridou, Michael

Loster, Thorsten Papenbrock, Julian Risch, Ahmad Samiei und Zhe Zuo.

Zwei weitere HPI-Fachgebiete, mit denen wir kooperieren, arbeiten ebenfalls in der Datenbank-Community: Das Fachgebiet „Enterprise Platforms and Integration Concepts" (EPIC) unter der Leitung von Hasso Plattner und Matthias Uflacker sowie das Fachgebiet „Knowledge Discovery und Data Mining" (KDD) unter der Leitung von Emmanuel Müller.

Literatur

1. Abedjan Z, Golab L Naumann F (2015) Profiling relational data: a survey. VLDB Journal, 24(4):557-581
2. Abedjan Z, Gruetze T, Jentzsch A, Naumann F (2014) Profiling and mining RDF data with ProLOD++. In Proc. of the International Conference on Data Engineering (ICDE), Demo. S 1198-1201
3. Agichtein E, Gravano L (2000) Snowball: Extracting relations from large plain-text collections. In Proc. of the ACM Conference on Digital Libraries. S 85-94
4. Agrawal D, Bernstein P, Bertino E, Davidson S, Dayal U, Franklin M, . . . Widom J (2012) Challenges and opportunities with Big Data. Technical report, Computing Community Consortium, http://cra.org/ccc/docs/init/bigdatawhitepaper.pdf
5. Draisbach U, Naumann F, Szott S, Wonneberg O (2012) Adaptive windows for duplicate detection. In Proc. of the International Conference on Data Engineering (ICDE). S 1073-1083
6. Godde C, Lazaridou K, Krestel R (2016) Classification of German Newspaper Comments. In Proc. of the Conference Lernen, Wissen, Daten, Analysen (LWDA). S 299-310
7. Gruetze T, Krestel R, Naumann F (2016) Topic Shifts in StackOverflow: Ask it like Socrates. In Proc. of the 21st International Conference on Applications of Natural Language to Information Systems (NLDB), volume 9612. S 213–221
8. Gruetze T, Yao G, Krestel R (2015) Learning temporal tagging behaviour. In Proc. of the Temporal Web Analytics Workshop (TempWeb) at the International World Wide Web Conference (WWW). S 1333-1338
9. Hernández MA, Stolfo SJ (1998) Real-world data is dirty: Data cleansing and the merge/purge problem. Data Mining and Knowledge Discovery, 2(1):9-37
10. Huhtala Y, Kärkkäinen J, Porkka P, Toivonen H (1999) TANE: An efficient algorithm for discovering functional and approximate dependencies. Computer Journal, 42(2):100-111
11. Köhler H, Link S, Zhou X (2015) Possible and certain SQL keys. In Proc. of the VLDB Endowment, 8(11):1118-1129
12. Krestel R, Wall A, Nejdl W (2012) Treehugger or Petrolhead? Identifying Bias by Comparing Online News Articles with Political Speeches. In Proc. of the International World Wide Web Conference (WWW). S 547-548
13. Krestel R, Werkmeister T, Wiradarma TP, Kasneci G (2015) Tweet-recommender: Finding relevant tweets for news articles. In Proc. of the International World Wide Web Conference (WWW). S 53-54
14. Kruse S, Jentzsch A, Papenbrock T, Kaoudi Z, Quiane-Ruiz JA, Naumann F (2016) RDFind: Scalable conditional inclusion dependency discovery in RDF datasets. In Proc. of the International Conference on Management of Data (SIGMOD). S 953-967
15. Lange D, Naumann F (2011) Efficient similarity search: Arbitrary similarity measures, arbitrary composition. In Proc of the International Conference on Information and Knowledge Management (CIKM). S 1679-1688
16. Papenbrock T, Bergmann T, Finke M, Zwiener J, Naumann F (2015) Data profiling with Metanome (demo). In Proc. of the VLDB Endowment, 8(12):1860-1871
17. Papenbrock T, Ehrlich J, Marten J, Neubert T, Rudolph JP, Schönberg M, Zwiener J, Naumann F (2015) Functional dependency discovery: An experimental evaluation of seven algorithms. Proceedings of the VLDB Endowment, 8(10):1082-1093
18. Papenbrock T, Naumann F (2017) A Hybrid Approach for Efficient Unique Column Combination Discovery. In Proc. der Fachtagung Business, Technologie und Web (BTW), accepted
19. Park J, Blume-Kohout M, Krestel R, Nalisnick E, Smyth P (2016) Analyzing NIH Funding Patterns over Time with Statistical Text Analysis. In Scholarly Big Data: AI Perspectives, Challenges, and Ideas, Workshop at AAAI. S 698-704
20. Weis M, Naumann F, Jehle U, Lufter J, Schuster H (2008) Industry-scale duplicate detection. In Proc. of the VLDB Endowment, 1(2):1253-1264
21. Zuo Z, Kasneci G, Gruetze T, Naumann F (2014) BEL: Bagging for entity linking. In Proc. of the International Conference on

Computational Linguistics (COLING). S
2075-2086